From Gödel to Lambek: Studies in the Foundations of Mathematics

Philip Scott
University of Ottawa

(Lambekfest, September 21, 2013
CRM, U. de Montréal)

Lambek has spent considerable time thinking about the foundations and philosophy of mathematics. I will focus today on some themes surrounding computability.

- What is a computable function?
- What are natural recursion theories?
- Can we reconcile the various philosophies of mathematics?
- What is truth and what are Gödel's Incompleteness Theorems?

What is a computable function?

Definition (Primitive Recursion)

The primitive recursive functions is the smallest class *Prim* of numerical functions generated by:
(i) Basic Functions (e.g. Zero function, Successor, Projection)
(ii) Closed under two rules:
- Composition
- Primitive Recursion: if $g(\vec{x}), h(\vec{x}, y, u) \in Prim$ then so is f:

$$f(\vec{x}, 0) = g(\vec{x}), \quad f(\vec{x}, S(y)) = h(\vec{x}, y, f(\vec{x}, y)) \ .$$

- Underlying idea goes back to Dedekind [1888].
- Primitive Recursive Arithmetic (PRA): Thoralf Skolem [1923].
- Further developed: R. Péter, Hilbert-Bernays, Goodstein.
- Functions in *Prim* were used by Gödel in his Incompleteness Theorem [1931]. (and there is Incompleteness even for PRA)

So, does Computable = Primitive Recursive?

Alas, no: Cantor's diagonal argument.

So what is a computable function? This was taken up in a remarkable development 1931–1937 (at Princeton, mostly).

- A. Church (1932-34) & students (Kleene, Rosser) developed (untyped) lambda calculus. Church Formulated *Church's Thesis (1936)*: the intuitively computable numerical functions are exactly those you can compute in λ-calculus. Originally not believed by Gödel.
- Kleene [1934-35] developed μ-recursive functions: add to *Prim* the scheme: we can form $f(\vec{x}) = \mu y.g(\vec{x}, y) = 0$ (for total functions, add the proviso: provided $\forall\vec{x}\exists y.g(\vec{x}, y) = 0$).
- Gödel-Herbrand [1934]: An equation calculus to define "computable" functions.

Church-Turing Thesis

- Turing [1936] introduced Turing machines: an abstract mechanical computing device. He gave a convincing analysis of the meaning of being "computable" with no restrictions on space or time. So now we had: *Church-Turing thesis*.

  Note: CT is not a mathematical statement: it is an experimental statement, identifying an intuitive class (= "computable" numerical functions) with a precise mathematical class of (partial) functions. Evidence???

- Church, Kleene, Turing [1936-37]: carefully proved the equivalence of the above models of computability: all notions give exactly the same class of computable functions! Gödel was now convinced of the truth of the Church-Turing thesis.

Some Newer Models of Computability

- E. Post [1943], A. A. Markov [1951]: computability based on string rewriting grammars. (cf. Thue [1914]). Can be proved equivalent to Turing computability.
- Another important period: 1960-61 (simultaneously and (almost) independently): *Unlimited Register Machines*.

  J. Lambek, Z. Melzak, M. Minsky, J. Shepherdson & J. Sturgis:

  Lambek's paper (*How to Program an Infinite Abacus*) was by far the simplest to read and used a highly graphical syntax.

Infinite Abacus

Locations: X, Y, Z, $\cdots$ (arbitrary capacity)

Counters: Unlimited supply of (indistinguishable) pebbles.

Elementary Instructions:

Start $\downarrow$	$\downarrow$ **Stop**	$\downarrow$ X^+ $\downarrow$	$\downarrow$ Y^- $\swarrow$ $\searrow$	(If $Y \neq \emptyset$, take one pebble away and go to the left; else go to right)

Programs are finite number of instructions, arranged in a flow chart (directed graph), with feedback loops.

Such flowcharts can be naturally represented in a free symmetric traced category with $\otimes$ = coproduct.

What are computable functions in categories?

When I first came to McGill as a postdoc in 1976-78, I asked Jim "shouldn't we learn about what the recursion theorists are doing?"

Jim said: No. We have our own natural notions of computation: the function(al)s computable in various free categories, e.g. in free cartesian categories, free ccc's, the free topos, etc. First let's do that, then we can compare.

Definition (Lawvere)

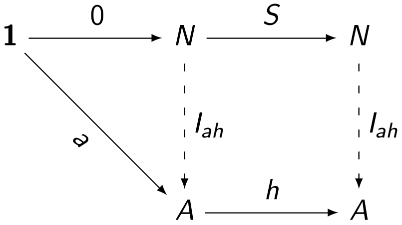
A Natural Numbers Object *(NNO) in a cartesian closed category is a diagram* $\mathbf{1} \xrightarrow{0} N \xrightarrow{S} N$ *initial among diagrams* $\mathbf{1} \xrightarrow{a} A \xrightarrow{h} A$*. i.e., there exists a unique* $I_{ah} : N \to A$ *satisfying:*

$$I_{ah}0 = a \quad , \quad I_{ah}S = hI_{ah}$$

Existence, without uniqueness, of I_{ah} *called* weak NNO

Primitive Recursion with parameters (weak NNO's)

Lawvere's NNO (for CCC's)

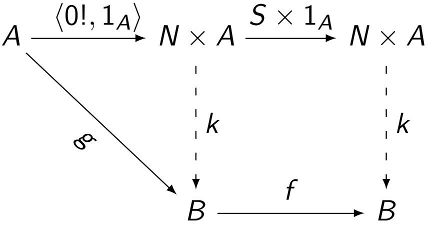


In Sets

$I_{ah}(0) = a$

$I_{ah}(n+1) = h(I_{ah}(n))$

Parametrized NNO: For $A \xrightarrow{g} B$, $B \xrightarrow{f} B$, $\exists \quad k : N \times A \to B$.



In Sets

$k(0, a) = g(a)$

$k(n+1, a) = f(k(n, a))$

Representing Numerical Functions in Categories

Definition (L-S, 1986)

Let $\mathcal{C}$ be a category with a weak NNO $\mathbf{1} \xrightarrow{0} N \xrightarrow{S} N$.
$f : \mathbb{N}^k \to \mathbb{N}$ *is* representable *in $\mathcal{C}$ if there is an arrow* $F : N^k \to N$ *such that* $F\langle \bar{n_1} \cdots \bar{n_k}\rangle = \overline{f(n_1, \cdots, n_k)}$, *where* $\bar{n} = S^n 0$.

A category is *Cartesian* if it has finite cartesian products.

Theorem (L. Roman, 1989)

The representable numerical functions in the free cartesian category $\mathcal{F}_c$ *with parametrized NNO are exactly Prim.*

Hence the unique representation functor $\mathcal{F}_c \longrightarrow$ **Set** has image $=$ the subcategory of sets with objects $=$ powers $\mathbb{N}^n$ and whose maps are tuples of primitive recursive functions.

Aside: NNO's in Monoidal Categories?

Cartesian categories, as deductive systems, correspond to the conjunction calculus $\mathcal{L} = \{\wedge, \top\}$. What about if we want to move to a substructural, or linear, logic $\mathcal{L} = \{\otimes, I\}$? (Cf. interesting paper, Paré-Roman: *Studia Logica*, 1989.)

Don't even assume symmetry (or permutation) $A \otimes B \xrightarrow{\sigma} B \otimes A$. Get *Left* and *Right* NNO's.

Definition (Left Parametrized NNO:)

For $A \xrightarrow{g} B, B \xrightarrow{f} B, \exists\, !\; k : N \otimes A \to B.$

$$\begin{array}{ccccc} I \otimes A & \xrightarrow{0 \otimes 1_A} & N \otimes A & \xrightarrow{S \otimes 1_A} & N \otimes A \\ \cong \downarrow & & \downarrow k & & \downarrow k \\ A & \xrightarrow{g} & B & \xrightarrow{f} & B \end{array}$$

Paré-Roman

Similarly for right NNO's. Although one might think projections (weakening) are necessary to define functions in *Prim*, that's not the case:

Theorem (Paré-Roman)

(i) The primitive recursive functions are representable in any monoidal category with LNNO.
(ii) Indeed, the free monoidal category with LNNO exists and is isomorphic to the free cartesian category with parametrized NNO.

Thus, again we get *Prim*.

How do we get more? Increase the logical strength (the types) from the logic of $\{\wedge, \top\}$ to $\{\wedge, \Rightarrow, \top\}$, as follows.

Free CCC with N

The free CCC with N is generated from simply typed lambda calculus, with iterators.

Objects: $A, B ::= \mathbf{1} \mid N \mid A \times B \mid A \Rightarrow B$

Arrows: generated by lambda terms. For example: typical terms

$$\begin{array}{cccc} \mathbf{1} & N & A \times B & A \Rightarrow B \\ * & 0 & \langle a, b \rangle & \lambda x : A.\varphi(x) \\ & Sn & & \end{array}$$

together with variables, evaluation $ev(f, a)$ (where $f : A \Rightarrow B$ and $a : A$) and iterators of every type $It_{a,h} : N \to A$.

Equations: add the minimal amount needed to have a CCC with weak NNO.

Representable Functions in the Free CCC with N

The following is a theorem in simply typed lambda calculus, translated into the language of ccc's by our equivalence (cf. earlier work by M-F Thibault, a student of Lambek):

Theorem (L-S, 1986)

In the free ccc $\mathcal{C}$ with weak NNO N, we have

1. *All primitive recursive functions and the Ackerman function are representable (in fact, if there's a strong NNO, by their usual free variable equations)*
2. *We represent a proper subclass of the total recursive functions.*

In fact, we used this to get an analog of Gödel's Incompleteness:

Theorem (A version of Incompleteness or 1 is not a generator)

There is a closed term $F : N \Rightarrow N$ such that for each numeral $\bar{n}$, $F\,`\bar{n} = \mathbf{0}$, but F is not the constant zero function, $\lambda x : N.0$.

Which functionals are representable in the free ccc?

We can identify arrows in the free ccc with N with closed lambda terms, modulo β, η, equivalently (by Curry-Howard-Lambek), as denoting proofs in $\{\wedge, \Rightarrow, \top\}$-logic.

- We can represent exactly the provably total functions of classical (first-order) Peano Arithmetic, i.e. those satisfying $\vdash \forall x \exists ! y A(x, y)$. These correspond to the ϵ_0-recursive functions, a proper subclass of the total recursive functions.
- The closed lambda terms represent a version of Gödel's Dialectica Functionals (= *the primitive recursive functionals of finite type*).
- They have been very influential in the Foundations of Constructive Analysis., e.g. Bishop, Troelstra, et. al. (most recently: Martín Escardó (MFPS'13) gave new proofs of continuity of functionals of type 2.)

Can we get still more numerical functions and functionals?

Higher Order Intuitionistic Logic: Full Type Theory

We can consider higher order intuitionistic logic with Peano's axioms: this is the internal language of toposes with NNO's.

Types:	1	Ω	N	Ω^A	$A \times B$
Terms:	$*$	$a = a'$ $a \in \alpha$	0 Sn	$\{x : A \mid \varphi(x)\}$	$\langle a, b\rangle$

In [LS86] we gave an axiomatization based on equality, comprehension, extensionality, and Peano's axioms. Following Russell, Henkin, and Prawitz, we base the logic on equality:

$$
\begin{array}{lcll}
\top & := & * = * & \\
p \wedge q & := & \langle p, q\rangle = \langle \top, \top\rangle & \text{where } \; p, q : \Omega \\
p \Rightarrow q & := & p \wedge q = p & \\
\forall_{x:A}\varphi(x) & := & \{x : A \mid \varphi(x)\} = \{x : A \mid \top\} & \text{where } \; \varphi(x) : \Omega \\
\bot & := & \forall_{x:\Omega} x & \\
\neg p & := & \forall_{x:\Omega}(p \Rightarrow x) &
\end{array}
$$

Higher Order Intuitionistic Logic: Full Type Theory

$$p \vee q \quad := \quad \forall_{x:\Omega}(((p \Rightarrow x) \wedge (q \Rightarrow x)) \Rightarrow x)$$

$$\exists_{x:A}\varphi(x) \quad := \quad \forall_{y:\Omega}(\forall_{x:A}((\varphi(x) \Rightarrow y) \Rightarrow y))$$

Definition (Generated Toposes $T(\mathcal{L})$)

The topos $T(\mathcal{L})$ generated by the type theory $\mathcal{L}$ has as objects "sets" (closed terms α of type Ω^A , modulo provable equality). Morphisms $\alpha \to \beta$, where $\alpha : \Omega^A$ and $\beta : \Omega^B$, are "provably functional relations", i.e. closed terms $\varphi : \Omega^{A\times B}$ (modulo provable equality) such that:

$$\vdash_{\mathcal{L}} \forall_{x:A}(x \in \alpha \Rightarrow \exists !_{y:B}(y \in \beta \wedge (x, y) \in \varphi))$$

$T(\mathcal{L})$ is the category of "sets" and "functions" formally definable in higher-order logic $\mathcal{L}$.

For $\mathcal{L}_0$ = pure type theory, $T(\mathcal{L}_0)$ is called the *free topos*.

Computable functions and the free topos

Lambek and I have spent a lot of time trying to convince people that the free topos is a perfect universe for moderate intuitionists.

Now the class of computable functions changes radically depending on whether we use classical or intuitionistic logic.

Theorem (LS86)

(i) The representable (= provably total) functions $N \to N$ in pure intuitionistic type theory are recursive.
(ii) Not all recursive functions so arise (by diagonalizing out of the class of all $\forall\exists!$-proofs.)

Question: is there a nice characterization of these functions?

The free Boolean Topos and classical type theory

The free Boolean topos is like the free topos, but generated from *classical* type theory. So we add the the law of excluded middle to pure type theory: $\forall p : \Omega(p \vee \neg p)$. (Alas, the free Boolean topos is *not* an ideal universe for classical mathematicians.)

Now the class of computable functions changes radically depending on whether we use classical or intuitionistic logic. Gödel first examined a weak form of representability (using numerals, instead of variables) in classical type theory.

Theorem (LS86)

(i) The numeral-wise representable functions in classical type theory are exactly the total recursive functions (Gödel).
(ii) The provably total functions of classical type theory coincide with the numeralwise representable ones (V. Huber-Dyson), when they define total functions.

Approaches to representing partial recursive functions

Unfortunately, the free Boolean topos has non-standard numerals $\mathbf{1} \xrightarrow{f} N$. This suggests maybe *partial functions* are better...

In [LS86] we observed that Gödel's notion of weak or numeralwise representability was perhaps more appropriate to the free topos:

Theorem

A partial numerical function is numeralwise representable in pure type theory (i.e. the free topos) iff it is partial recursive.

There are other, more interesting recent categorical approaches to partial recursive functions via partial map (restriction- and Turing-) categories (Cockett-Hofstra) but we haven't yet developed that work here.

But I'd like to mention a recent, relevant talk of Plotkin (for the Abramsky fest), since this also uses a Lambek Lemma.

Plotkin: another use of Lambek's Lemma

Recall Lambek's Lemma, often used in Domain Theory (cf. work of Peter Freyd.) One has a functor $T : \mathcal{C} \to \mathcal{C}$ and we define a T *algebra* to be a map $TA \to A$. We may form the category $T - Alg$ of T-algebras, by having maps = commutative squares. An *initial* T-algebra is one which has a unique map to any other T-algebra.

Lemma (Lambek)

If $TA \xrightarrow{f} A$ is an initial T-algebra, then f is an isomorphism.

$\mathbb{N}$ is an NNO in Sets (following Lawvere). We look at the functor $T(-) = \mathbf{1} + (-)$. A *natural number object* is an initial algebra for this: a map $TN \to N$ = a map $(\mathbf{1} + N) \to N$ is a *pair* of maps: $\mathbf{1} \xrightarrow{0} N$ and $N \xrightarrow{S} N$, initial amongst all such diagrams.

Plotkin's Theorem

In **Sets**, Lambek's Lemma would say the familiar fact:
$\mathbf{1} + \mathbb{N} \xrightarrow{\alpha} \mathbb{N}$ is an iso , for $\alpha = [0, S]$.

As we know: *Initiality* of α gives us primitive recursion. What about if we turn things around, and ask for the *finality* $\alpha^{-1} : \mathbb{N} \to \mathbf{1} + \mathbb{N}$? It turns out this gives exactly Kleene μ-recursion for partial functions.

Plotkin's Theorem

Theorem (Plotkin,2013)

Let $\mathcal{C}$ be a monoidal category with (right distributive) binary sums and a weak left (or right) natural numbers object $I \xrightarrow{0} N \xrightarrow{S} N$ such that $[0, S]$ is an isomorphism and $(N, [0, S]^{-1})$ is a weakly final natural numbers coalgebra. Then all partial recursive functions with recursive graphs are strongly representable in $\mathcal{C}$ (assuming that $S \circ c \neq 0$, for all $c : I \to N$).

I find the above more pleasing than moving to *Untyped* systems, like untyped lambda calculi.

What are some different philosophies of mathematics?

Lambek has always been interested in foundations of mathematics. There are various famous philosophies of mathematics:

Logicism: (Frege) the primacy of symbolic logic: mathematics is reduced to logic. Numbers can be defined in terms of logic (e.g. Russell).

Platonism: basic mathematical objects exist, independently of the human mind. E.g. numbers are abstract, necessarily existing.

Intuitionism: (Brouwer) Proofs and mathematical objects are mental constructions. Aristotle's law of excluded middle is not valid.

Formalism: (Hilbert) Mathematics can be considered as a formal game with symbols, manipulated according to certain rules.

Constructive Nominalism?

Lambek-Couture[1991] introduced this notion to discuss merging "moderate" adherents of the above philosophies. Lambek and I (in [LS86] and our recent work) and Lambek in many recent papers, have been pushing the idea:

The free topos is an ideal universe for a moderate intuitionist, which also encodes many aspects of the previous philosophies:

– Platonists, because as an initial object it is unique up to isomorphism;

– Formalists, or even nominalists, because of its linguistic construction;

– Constructivists, or moderate intuitionists, because the underlying type theory is intuitionistic and it satisfies numerous constructive principles.

Constructive Nominalism?

– Logicists, because this type theory is a form of higher order logic, although it must be complemented by an axiom of infinity, say in the form of Peano's axioms.

Although there might be some objections:

- Unlike the pure logicists, we adjoin the Peano Laws.
- While most mathematicians accept impredicative type theory as legitimate, not followers of Martin-Löf's type theories (e.g. recent work on univalent foundations & homotopy lambda calculus, by Voevodsky, Awodey, Coquand, et. al.)
- Many technical intuitionists want *laws* or *axioms*, whereas the free topos is only closed under *rules*: e.g. the *Rule of Choice*. (Alas, the full Axiom of Choice $\Rightarrow$ Booleaness).

Some properties of the free topos/pure type theory

Pure intuitionistic type theory $\mathcal{L}_0$ has many interesting properties, which translate into algebraic properties of the free topos:

Consistency: not ($\vdash \perp$) .

Disjunction Property: If $\vdash_{\mathcal{L}_0} p \vee q$, then $\vdash_{\mathcal{L}_0} p$ or $\vdash_{\mathcal{L}_0} q$.

Existence Property: If $\vdash_{\mathcal{L}_0} \exists_{x:A}\varphi(x)$ then $\vdash_{\mathcal{L}_0} \varphi(a)$ for some closed term a of type A.

Troelstra's Uniformity principle for $A = \Omega^C$: If $\vdash_{\mathcal{L}_0} \forall_{x:A}\exists_{y:N}\varphi(x,y)$ then $\vdash_{\mathcal{L}_0} \exists_{y:N}\forall_{x:A}\varphi(x,y)$. In the free topos $\mathcal{F}$, the uniformity principle says the arrows $\Omega^C \to N$ are constant.

Independence of premisses: If $\vdash_{\mathcal{L}_0} \neg p \Rightarrow \exists_{x:A}\varphi(x)$ then $\vdash_{\mathcal{L}_0} \exists_{x:A}(\neg p \Rightarrow \varphi(x))$.

Markov's Rule: If $\vdash_{\mathcal{L}_0} \forall_{x:A}(\varphi(x) \vee \neg\varphi(x))$ and $\vdash_{\mathcal{L}_0} \neg\forall_{x:A}\neg\varphi(x)$, then $\vdash_{\mathcal{L}_0} \exists_{x:A}\varphi(x)$.

The Existence Property with a parameter of type $A = \Omega^C$: if $\vdash_{\mathcal{L}_0} \forall_{x:A}\exists_{y:B}\varphi(x,y)$ then $\vdash_{\mathcal{L}_0} \forall_{x:A}\varphi(x,\psi(x))$, where $\psi(x)$ is some term of type B.

What is Gödel's Incompleteness Theorem?

Jim has been thinking about Incompleteness Theorems for quite some time. Two publications include:

(i) A proof in his book with W. Anglin, *The Heritage of Thales*.

(ii) L-S (2011): Reflections on a categorical foundations of mathematics.

I want to talk a bit about some of Lambek's recent ideas.

ω-Rules and truth

One very interesting concept Lambek recently thought about is the ω-rule (semantic form) (here we stick to models whose only terms $t : N$ are numerals).

$$\frac{A(\bar{n}) \ \ \textbf{true}, \ \text{for all } n \in \mathbb{N}}{\forall x : N.A(x) \ \ \textbf{true}}$$

Classically, note this rule is equivalent to a semantic version of the existence property:

$$\frac{\exists x : N.B(x) \ \ \textbf{true}}{B(\bar{n}) \ \ \textbf{true}, \ \text{for some } n \in \mathbb{N}}$$

Of course, these are not intuitionistically equivalent (formally). Suppose we want to analyze this in a deeper way.

A famous question of Pontius Pilate: "What is truth?"

What is truth in intuitionism?

Interesting paper: "Conceptions of truth in intuitionism", P. Raatikainen, *History & Philosophy of Logic*, 2004.

Examines carefully many published viewpoints (from Brouwer, Heyting, to Dummett). "It is argued that each account faces difficult problems. They all either have implausible consequences or are viciously circular"

1. (Summarizing many quotes of Brouwer): a proposition is true only if it has been actually proved, and an object exists only if it has been actually constructed.
2. Brouwer sometimes equates truth with provability in principle. At first he was obsessed with absolutely unsolvable problems, but later in 1950's liberalized his view.

What is truth in intuitionism?

3 Brouwer (and Heyting) oscillated between two views (called 'actualist' vs. 'possibilist'): does truth = *actual possession of a proof* or does truth = *possibility of constructing a proof*.

4 Dummett (1982): it seems better to represent a constructivist theory of meaning for mathematical statements as dispensing with notion of truth altogether.

Anyway, suppose we accept, following Lambek, intuitionistic truth = "knowable" = "provable".

Then for the classically equivalent ω-rules (for truth), things split: the second rule becomes the $\exists$-property. And the latter is actually *a derivable rule* (Lambek and I gave three proofs of $\exists$- and $\vee$-properties for intuitionist HOL.)

What is truth in intuitionism?

But (even for classical logic) the first ω-rule (for provability) fails, as we saw for free ccc:

$$\frac{\vdash A(\bar{n}) \quad \text{for all } n \in \mathbb{N}}{\vdash \forall x : N.A(x)}$$

This is a consequence of the proof of Gödel's Incompleteness Theorem.

Let's sketch this now.

Gödel's Incompleteness Theorem

Suppose we look at models satisfying the semantic ω-rule.

Theorem (Gödel)

Suppose $\mathcal{L}_0$ is consistent. Then there is a formula G true in any model satisfying the ω-rule, but not provable.

Proof: Enumerate the sets (the terms of type Ω^N or $P(N)$), which are comprehension terms $\{x : N \mid A(x)\}$. Call them $\alpha_1, \alpha_2, \cdots$ ..
Enumerate the formal proofs in $\mathcal{L}_0$: $\pi_1, \pi_2, \cdots$.
Let $R(m,n) := \pi_n$ proves $\bar{m} \in \alpha_m$.
This is a recursive predicate. Gödel showed us how to represent this formally in the language. So there is a formula F such that

$$\begin{array}{rcl} R(m,n) & \Rightarrow & \vdash F(\bar{m},\bar{n}) \\ \neg R(m,n) & \Rightarrow & \vdash \neg F(\bar{m},\bar{n}) \end{array}$$

Let $\alpha_g := \{x : N \mid \neg\exists y : N . F(x,y)\}$.

Gödel's Incompleteness Theorem II

Let $\alpha_g := \{x : N \mid \neg\exists y : N.F(x,y)\}$. So

$$\vdash \bar{g} \in \alpha_g \leftrightarrow \neg\exists y : N.F(\bar{g}, y)\}.$$

Let $G = \bar{g} \in \alpha_g$. Suppose $\vdash G$. Then for some n, π_n proves G. So $R(g,n)$. So $\vdash F(\bar{g}, \bar{n})$. So $\vdash \exists y : N.F(\bar{g}, y)$. Therefore, $\vdash \neg\neg\exists y : N.F(\bar{g}, y)$, even intuitionistically.
So even intuitionistically, $\vdash \neg G$. Contradiction, since $\mathcal{L}$ is consistent.

So, $\not\vdash G$. Hence, for all n, π_n is not a proof of G. so $\forall n \in \mathbb{N}, \neg R(g,n)$. Hence $\forall n \in \mathbb{N}, \vdash \neg F(\bar{g}, \bar{n})$ Exercise: $\forall y : N \neg F(\bar{g}, y)$ is not provable, even intuitionistically. It is, however, true.

With a technical condition (ω-consistency), one can even prove also $\not\vdash \neg\forall y.\neg F(\bar{g}, y)$. Hence $\not\vdash \neg G$ also. So neither G nor $\neg G$ is provable.

Gödel's Incompleteness Theorem III

Jim and I understood a model to be a non-trivial topos satisfying the disjunction and existence property (1 is an indecomposable projective). By an *analytic type theory* we mean one which is a quotient of the free language $\mathcal{L}_0$. The following is true for both intuitionistic and classical systems.

Theorem (LS,2011)

If $\mathcal{L}$ is a consistent analytic type theory whose theorems are r.e., there is a proposition q satisfying: if $\mathcal{L}$ has at least one model in which the numerals are standard, then neither q nor $\neg q$ is a theorem.

Has many corollaries, e.g. the set of true statements of $\mathcal{L}_0$ (in those Boolean toposes in which 1 is a generator and in which numerals are standard) cannot be r.e.